

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号
特開2001-147864
(P2001-147864A)

(43)公開日 平成13年5月29日(2001.5.29)

(51)Int.Cl. ⁷	識別記号	F I	テーマコード(参考)
G 0 6 F 12/16	3 4 0	G 0 6 F 12/16	3 4 0 P 5 B 0 1 8
12/00	5 6 0	12/00	5 6 0 A 5 B 0 6 0
12/02	5 1 0	12/02	5 1 0 A

審査請求 未請求 請求項の数7 O L (全 18 頁)

(21)出願番号 特願平11-329986

(22)出願日 平成11年11月19日(1999.11.19)

(71)出願人 000002369

セイコーエプソン株式会社
東京都新宿区西新宿2丁目4番1号

(72)発明者 旭 常盛

長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内

(74)代理人 100093388

弁理士 鈴木 喜三郎 (外2名)

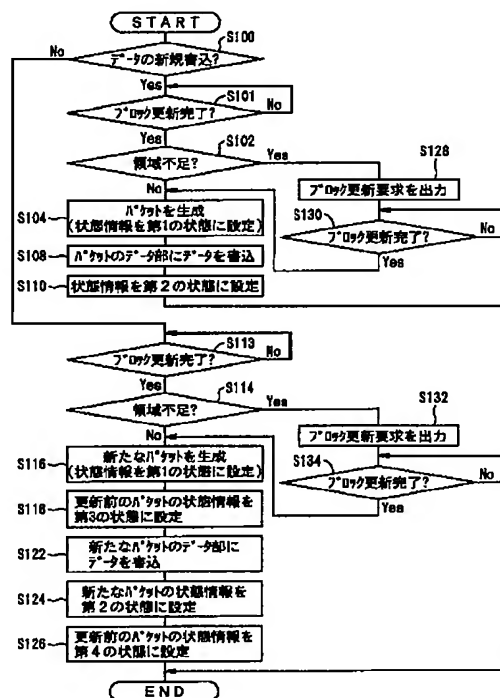
Fターム(参考) 5B018 GA06 HA23 KA03 KA30 NA06
QA05 RA11
5B060 AA03 AA06 AA14 AC11 CA14

(54)【発明の名称】 フラッシュメモリのデータ管理装置

(57)【要約】

【課題】 フラッシュメモリの寿命の低下を防止するとともにデータの書換を高速に行い、しかもデータの更新を高い信頼性をもって行うのに好適なフラッシュメモリのデータ管理装置を提供する。

【解決手段】 パケットのデータを更新するときは、更新すべきデータを格納するための新たなパケットをパケット部72に生成し、データ書込開始時にパケットの状態情報を第3の状態に設定するとともに新たなパケットの状態情報を第1の状態に設定し、新たなパケットのデータ部82に更新すべきデータを書き込み、データ書込完了時に新たなパケットの状態情報を第2の状態に設定するとともにパケットの状態情報を第4の状態に設定し、状態情報が第2の状態または第3の状態であるパケットを有効なパケットであると判定し、状態情報が第1の状態または第4の状態であるパケットを無効なパケットであると判定する。



【特許請求の範囲】

【請求項1】 フラッシュメモリの記憶領域にパケット単位でデータを記憶する装置であって、前記パケットは、当該パケットの管理に関する管理データを格納するための管理部と、記憶すべきデータを格納するためのデータ部とからなり、前記管理部には、前記管理データとして、前記パケットの異なる2つの状態を示す状態情報が格納可能となっており、データを記憶するときは、記憶すべきデータを格納するためのパケットを前記記憶領域に生成し、データ書込開始時に前記パケットの状態情報をデータの新規書込中であることを示す第1の状態に設定し、前記パケットのデータ部に前記記憶すべきデータを書き込み、データ書込完了時に前記パケットの状態情報をデータの新規書込が完了したことを示す第2の状態に設定し、前記状態情報が前記第2の状態であるパケットを有効なパケットであると判定し、前記状態情報が前記第1の状態であるパケットを無効なパケットであると判定するようになっていることを特徴とするフラッシュメモリのデータ管理装置。

【請求項2】 請求項1において、前記状態情報は、前記パケットのさらに異なる2つの状態を示すようになっており、前記パケットのデータを更新するときは、更新すべきデータを格納するための新たなパケットを前記記憶領域に生成し、データ書込開始時に前記パケットの状態情報をデータの更新中であることを示す第3の状態に且つ前記新たなパケットの状態情報を前記第1の状態に設定し、前記新たなパケットのデータ部に前記更新すべきデータを書き込み、データ書込完了時に前記新たなパケットの状態情報を前記第2の状態に且つ前記パケットの状態情報をデータの更新が完了したことを示す第4の状態に設定し、前記状態情報が前記第2の状態又は前記第3の状態であるパケットを有効なパケットであると判定し、前記状態情報が前記第1の状態又は前記第4の状態であるパケットを無効なパケットであると判定するようになっていることを特徴とするフラッシュメモリのデータ管理装置。

【請求項3】 請求項1及び2のいずれかにおいて、前記パケットは、可変長であり、前記管理部には、前記管理データとして、当該パケットのデータ長を示すデータ長情報が格納可能となっており、前記データ長情報に基づいて、前記パケットの格納位置を特定するようになっていることを特徴とするフラッシュメモリのデータ管理装置。

【請求項4】 請求項1乃至3のいずれかに記載のフラッシュメモリのデータ管理装置により前記記憶領域にパケット単位でデータを記憶し、前記記憶領域とは異なる

別の記憶領域を用いて当該記憶領域を更新する装置であって、

前記記憶領域は、当該記憶領域の管理に関する領域管理データを格納するための領域管理部と、パケットを格納するためのパケット部とからなり、

前記領域管理部には、前記領域管理データとして、前記記憶領域の異なる4つの状態を示す領域状態情報が格納可能となっており、

前記記憶領域を更新するときは、データ書込開始時に前記記憶領域の領域状態情報をデータの更新中であることを示す第3の状態に且つ前記別の記憶領域の領域状態情報をデータの新規書込中であることを示す第1の状態に設定し、前記記憶領域の有効なパケットのデータを前記別の記憶領域のパケット部書き込み、データ書込完了時に前記別の記憶領域の領域状態情報をデータの新規書込が完了したことを示す第2の状態に且つ前記記憶領域の領域状態情報をデータの更新が完了したことを示す第4の状態に設定し、

前記領域状態情報が前記第2の状態又は前記第3の状態である記憶領域を有効な記憶領域であると判定し、前記領域状態情報が前記第1の状態又は前記第4の状態である記憶領域を無効な記憶領域であると判定するようになっていることを特徴とするフラッシュメモリのデータ管理装置。

【請求項5】 請求項4において、前記領域状態情報が前記第1の状態又は前記第4の状態である記憶領域を前記別の記憶領域として用いるようになっていることを特徴とするフラッシュメモリのデータ管理装置。

【請求項6】 請求項4及び5のいずれかにおいて、前記記憶領域の領域状態情報及び前記別の記憶領域の領域状態情報の状態に応じて前記記憶領域又は前記別の記憶領域に対する処理を割り当てた処理定義テーブルを参照して、前記記憶領域又は前記別の記憶領域に対する処理を実行するようになっていることを特徴とするフラッシュメモリのデータ管理装置。

【請求項7】 請求項1乃至6のいずれかにおいて、前記記憶領域は、フラッシュメモリのデータ初期化単位からなることを特徴とするフラッシュメモリのデータ管理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、フラッシュメモリの記憶領域にパケット単位でデータを記憶する装置に係り、特に、フラッシュメモリのデータ初期化単位からなる記憶領域に可変長のパケット単位でデータを記憶する装置に関する。さらに詳しくは、フラッシュメモリの初期化回数を少なくして、フラッシュメモリの寿命の低下を防止するとともにデータの書換を高速に行い、しかもデータの更新を高い信頼性をもって行うのに好適なフラ

ッシュメモリのデータ管理装置に関する。

【0002】

【従来の技術】従来、フラッシュメモリの記憶領域にパケット単位でデータを記憶する装置としては、例えば、特開平5-233426号公報に開示されたフラッシュメモリの使用方法がある。これは、フラッシュメモリを複数のセクタで区切り、各セクタに論理アドレス部分と、消去管理表示部分と、データ部分とを設け、論理アドレスによりセクタを検索する。そして更新するとき、セクタの消去管理表示部分の消去可能フラグをオンにし、空きセクタにその論理アドレスを付加し、更新データを書き込む。

【0003】これにより、フラッシュメモリ上でのデータの消去は、消去対象となるデータが格納されているセクタの消去管理部にデータが無効であることを示す無効フラグを設定することにより行われるので、フラッシュメモリの初期化回数が少なくなり、フラッシュメモリの寿命の低下を防止するとともにデータの書換が高速に行える。

【0004】

【発明が解決しようとする課題】しかしながら、上記従来のフラッシュメモリの使用方法にあつては、セクタのデータを更新する場合は、更新すべきデータを新たなセクタに書き込み、更新前のセクタの消去管理部に無効フラグを設定するため、例えば、停電やシステムのハングアップまたはリセットスイッチの誤投入といった事態がデータの更新中に生じた場合は、更新前のセクタと新たなセクタの消去管理部にデータが有効であることを示す有効フラグがともに設定されたまま、データの更新が中断されてしまい、どちらが正しいデータであるかを識別することが困難であるという問題があった。

【0005】こうした問題に対処するためには、更新前のセクタが属するブロックを一旦初期化し、その上で、更新すべきデータを新たなセクタに書き込むということをしなければならず、結局、フラッシュメモリの初期化回数を少なくすることができない。なお、この場合は、初期化したブロックのうち更新前のセクタ以外のものを、元の状態に書き換え直す必要がある。

【0006】そこで、本発明は、このような従来の技術の有する未解決の課題に着目してなされたものであつて、フラッシュメモリの初期化回数を少なくして、フラッシュメモリの寿命の低下を防止するとともにデータの書換を高速に行い、しかもデータの更新を高い信頼性をもって行うのに好適なフラッシュメモリのデータ管理装置を提供することを目的としている。

【0007】

【課題を解決するための手段】上記目的を達成するために、本発明に係る請求項1記載のフラッシュメモリのデータ管理装置は、フラッシュメモリの記憶領域にパケット単位でデータを記憶する装置であつて、前記パケット

は、当該パケットの管理に関する管理データを格納するための管理部と、記憶すべきデータを格納するためのデータ部とからなり、前記管理部には、前記管理データとして、前記パケットの異なる2つの状態を示す状態情報が格納可能となっており、データを記憶するときは、記憶すべきデータを格納するためのパケットを前記記憶領域に生成し、データ書込開始時に前記パケットの状態情報をデータの新規書込中であることを示す第1の状態に設定し、前記パケットのデータ部に前記記憶すべきデータを書き込み、データ書込完了時に前記パケットの状態情報をデータの新規書込が完了したことを示す第2の状態に設定し、前記状態情報が前記第2の状態であるパケットを有効なパケットであると判定し、前記状態情報が前記第1の状態であるパケットを無効なパケットであると判定するようになっている。

【0008】このような構成であれば、データを記憶するときは、まず、記憶すべきデータを格納するためのパケットが記憶領域に生成され、データ書込開始時には、パケットの状態情報が第1の状態に設定される。そして、パケットのデータ部に記憶すべきデータが書き込まれ、データ書込完了時には、パケットの状態情報が第2の状態に設定される。

【0009】例えば、何らかの原因によりデータの書込が中断された場合は、状態情報が第2の状態であるパケットが有効なパケットであると判定され、状態情報が第1の状態であるパケットが無効なパケットであると判定される。

【0010】したがって、データの書込が完了するまでは、有効なパケットとして識別されない。例えば、データを更新する場合は、更新すべきデータが新たなパケットに書き込まれるが、新たなパケットはデータの書込が完了するまでは、有効なパケットとして識別されないのので、データの書込が完了するまでの間に何らかの原因によりデータの書込が中断されても、更新前のパケットの方が正しいデータであると識別される。

【0011】ここで、パケットは、どのようなデータ長であってもよく、固定長であってもよいし、可変長であってもよい。

【0012】さらに、本発明に係る請求項2記載のフラッシュメモリのデータ管理装置は、請求項1記載のフラッシュメモリのデータ管理装置において、前記状態情報は、前記パケットのさらに異なる2つの状態を示すようになつており、前記パケットのデータを更新するときは、更新すべきデータを格納するための新たなパケットを前記記憶領域に生成し、データ書込開始時に前記パケットの状態情報をデータの更新中であることを示す第3の状態にかつ前記新たなパケットの状態情報を前記第1の状態に設定し、前記新たなパケットのデータ部に前記更新すべきデータを書き込み、データ書込完了時に前記新たなパケットの状態情報を前記第2の状態にかつ前記

パケットの状態情報をデータの更新が完了したことを示す第4の状態に設定し、前記状態情報が前記第2の状態または前記第3の状態であるパケットを有効なパケットであると判定し、前記状態情報が前記第1の状態または前記第4の状態であるパケットを無効なパケットであると判定するようになっている。

【0013】このような構成であれば、パケットのデータを更新するときは、まず、更新すべきデータを格納するための新たなパケットが記憶領域に生成され、データ書込開始時には、更新前のパケットの状態情報が第3の状態にかつ新たなパケットの状態情報が第1の状態に設定される。そして、新たなパケットのデータ部に更新すべきデータが書き込まれ、データ書込完了時には、新たなパケットの状態情報が第2の状態にかつ更新前のパケットの状態情報が第4の状態に設定される。

【0014】例えば、何らかの原因によりデータの更新が中断された場合は、状態情報が第2の状態または第3の状態であるパケットが有効なパケットであると判定され、状態情報が第1の状態または第4の状態であるパケットが無効なパケットであると判定される。

【0015】したがって、データの更新が完了するまでは、新たなパケットは、有効なパケットとして識別されない。

【0016】ここで、新たなパケットの状態情報を第2の状態にかつ更新前のパケットの状態情報を第4の状態に設定する順番は、どちらが先であってもよいが、新たなパケットの状態情報を第2の状態に設定してから、更新前のパケットの状態情報を第4の状態に設定する順番の方が好ましい。これは、新たなパケットの状態情報を変更した後、更新前のパケットの状態情報を変更する前に、何らかの原因によりデータの更新が中断された場合でも、更新前のパケットが有効なパケットであると判定されるからである。

【0017】また、パケットの状態情報をデータの更新中であることを示す第3の状態にかつ新たなパケットの状態情報を第1の状態に設定する順番は、どちらが先であってもよい。

【0018】さらに、本発明に係る請求項3記載のフラッシュメモリのデータ管理装置は、請求項1および2のいずれかに記載のフラッシュメモリのデータ管理装置において、前記パケットは、可変長であり、前記管理部には、前記管理データとして、当該パケットのデータ長を示すデータ長情報が格納可能となっており、前記データ長情報に基づいて、前記パケットの格納位置を特定するようになっている。

【0019】このような構成であれば、記憶領域には、可変長のパケットが記憶され、パケットの格納位置は、データ長情報に基づいて特定される。

【0020】さらに、本発明に係る請求項4記載のフラッシュメモリのデータ管理装置は、請求項1ないし3の

いずれかに記載のフラッシュメモリのデータ管理装置により前記記憶領域にパケット単位でデータを記憶し、前記記憶領域とは異なる別の記憶領域を用いて当該記憶領域を更新する装置であって、前記記憶領域は、当該記憶領域の管理に関する領域管理データを格納するための領域管理部と、パケットを格納するためのパケット部とからなり、前記領域管理部には、前記領域管理データとして、前記記憶領域の異なる4つの状態を示す領域状態情報が格納可能となっており、前記記憶領域を更新するときは、データ書込開始時に前記記憶領域の領域状態情報をデータの更新中であることを示す第3の状態にかつ前記別の記憶領域の領域状態情報をデータの新規書込中であることを示す第1の状態に設定し、前記記憶領域の有効なパケットのデータを前記別の記憶領域のパケット部に書き込み、データ書込完了時に前記別の記憶領域の領域状態情報をデータの新規書込が完了したことを示す第2の状態にかつ前記記憶領域の領域状態情報をデータの更新が完了したことを示す第4の状態に設定し、前記領域状態情報が前記第2の状態または前記第3の状態である記憶領域を有効な記憶領域であると判定し、前記領域状態情報が前記第1の状態または前記第4の状態である記憶領域を無効な記憶領域であると判定するようになっている。

【0021】このような構成であれば、記憶領域にデータを書き込む領域がなくなった等の理由により、記憶領域を更新するときは、データ書込開始時には、更新前の記憶領域の領域状態情報が第3の状態にかつ更新先となる別の記憶領域の領域状態情報が第1の状態に設定される。そして、更新前の記憶領域の有効なパケットのデータが別の記憶領域のパケット部に書き込まれ、データ書込完了時には、別の記憶領域の領域状態情報が第2の状態にかつ更新前の記憶領域の領域状態情報が第4の状態に設定される。

【0022】例えば、何らかの原因により記憶領域の更新が中断された場合は、状態情報が第2の状態または第3の状態である記憶領域が有効な記憶領域であると判定され、状態情報が第1の状態または第4の状態である記憶領域が無効なパケットであると判定される。

【0023】したがって、記憶領域の更新が完了するまでは、更新先となる別の記憶領域は、有効な記憶領域として識別されない。

【0024】ここで、フラッシュメモリが複数の記憶領域からなる場合は、更新先となる別の記憶領域は、それら記憶領域のうち記憶領域の更新を同時に行う数だけあればよい。記憶領域の更新を同時に行う数は、システムの設計内容によるが、一般的には、記憶領域の更新を同時に複数行わないので、更新先となる別の記憶領域は、フラッシュメモリの記憶領域のうち一つあれば足りる。

【0025】また、記憶領域を更新するときは、記憶領域の有効なパケットのデータを別の記憶領域のパケット

部に書き込むようになっているが、別の記憶領域を有効に利用する観点からは、記憶領域の有効なパケットのデータを、その配置位置が整順されるように、別の記憶領域のパケット部に書き込むようになっているのが好ましい。

【0026】ここで、別の記憶領域の領域状態情報を第2の状態にかつ記憶領域の領域状態情報を第4の状態に設定する順番は、どちらが先であってもよい。また、記憶領域の領域状態情報を第3の状態にかつ別の記憶領域の領域状態情報を第1の状態に設定する順番は、どちらが先であってもよい。

【0027】さらに、本発明に係る請求項5記載のフラッシュメモリのデータ管理装置は、請求項4記載のフラッシュメモリのデータ管理装置において、前記領域状態情報が前記第1の状態または前記第4の状態である記憶領域を前記別の記憶領域として用いるようになっている。

【0028】このような構成であれば、記憶領域の更新が少なくとも1回行われた後は、領域状態情報が第1の状態または第4の状態である記憶領域が別の記憶領域として用いられる。

【0029】さらに、本発明に係る請求項6記載のフラッシュメモリのデータ管理装置は、請求項4および5のいずれかに記載のフラッシュメモリのデータ管理装置において、前記記憶領域の領域状態情報および前記別の記憶領域の領域状態情報の状態に応じて前記記憶領域または前記別の記憶領域に対する処理を割り当てた処理定義テーブルを参照して、前記記憶領域または前記別の記憶領域に対する処理を実行するようになっている。

【0030】このような構成であれば、処理定義テーブルを参照して、更新前の記憶領域または更新先となる別の記憶領域に対する処理が実行される。

【0031】ここで、処理定義テーブルの内容は、具体的に次のようになる。更新前の記憶領域の領域状態情報が第4の状態でありかつ更新先となる別の記憶領域の領域状態情報が第1の状態であるときは、更新前の記憶領域の有効なパケットのデータを別の記憶領域のパケット部に書き込む。更新前の記憶領域の領域状態情報が第4の状態でありかつ別の記憶領域の領域状態情報が第2の状態であるときは、更新前の記憶領域を初期化する。なお、更新前の記憶領域の領域状態情報と別の記憶領域の領域状態情報とが逆の場合は、記憶領域または別の記憶領域に対する処理も逆になる。

【0032】また、領域状態情報は、記憶領域の異なる4つの状態を示すようになっているが、記憶領域のさらに異なる2つの状態、すなわち、記憶領域が初期化されていないことを示す第5の状態、および記憶領域が初期化されていることを示す第6の状態を示すようにしておけば、処理定義テーブルの内容は、具体的に次のようになる。

【0033】更新前の記憶領域の領域状態情報が第3の状態でありかつ更新先となる別の記憶領域の領域状態情報が第1の状態であるときは、別の記憶領域を初期化する。更新前の記憶領域の領域状態情報が第4の状態でありかつ別の記憶領域の領域状態情報が第2の状態であるときは、更新前の記憶領域を初期化する。更新前の記憶領域の領域状態情報が第3の状態でありかつ別の記憶領域の領域状態情報が第5の状態であるときは、別の記憶領域を初期化する。更新前の記憶領域の領域状態情報が第3の状態でありかつ別の記憶領域の領域状態情報が第6の状態であるときは、更新前の記憶領域の有効なパケットのデータを別の記憶領域のパケット部に書き込む。なお、更新前の記憶領域の領域状態情報と別の記憶領域の領域状態情報とが逆の場合は、記憶領域または別の記憶領域に対する処理も逆になる。

【0034】このような構成であれば、更新前の記憶領域の初期化中、別の記憶領域の更新中、および別の記憶領域へのデータ書込中のいずれかに処理が中断されても、システムを再起動すると、中断時から行われるべき処理が実行されるので、記憶領域の更新が適切に行われる。したがって、データの更新をさらに高い信頼性をもって行うことができる。

【0035】さらに、本発明に係る請求項7記載のフラッシュメモリのデータ管理装置は、請求項1ないし6のいずれかに記載のフラッシュメモリのデータ管理装置において、前記記憶領域は、フラッシュメモリのデータ初期化単位からなる。

【0036】このような構成であれば、フラッシュメモリの記憶領域にパケット単位でデータを記憶・更新するときは、データ初期化単位からなる記憶領域内でそれらが行われる。また、記憶領域を更新するときは、データ初期化単位からなる記憶領域ごとにそれが行われる。

【0037】

【発明の実施の形態】以下、本発明の実施の形態を図面を参照しながら説明する。図1ないし図13は、本発明に係るフラッシュメモリのデータ管理装置の実施の形態を示す図である。

【0038】この実施の形態は、本発明に係るフラッシュメモリのデータ管理装置を、図1に示すように、フラッシュメモリであるフラッシュROM52のデータ初期化単位からなるブロックに、コンピュータ100の設定に必要なデータ（例えば、IPアドレス、アクセスすべきWWW（World Wide Web）サーバのURL（Universal Resource Locator））を、可変長のパケット単位で記憶する場合について適用したものである。

【0039】まず、本発明に係るフラッシュメモリのデータ管理装置を適用するコンピュータの構成を図1を参照しながら説明する。図1は、コンピュータ100の構成を示すブロック図である。

【0040】コンピュータ100は、図2に示すよう

に、制御プログラムに基づいて演算およびシステム全体を制御するCPU50と、所定領域にあらかじめCPU50の制御プログラム等を格納しているフラッシュROM52と、フラッシュROM52等から読み出したデータやCPU50の演算過程に必要な演算結果を格納するためのRAM54と、VRAM55に格納されているデータを画像信号に変換して出力するグラフィックチップ56と、外部装置に対してデータの入出力を媒介するI/F58とで構成されており、これらは、データを転送するための信号線であるバス59で相互にかつデータ授受可能に接続されている。

【0041】I/F58には、外部装置として、ヒューマンインターフェースとしてデータの入力が可能でかつ画像信号に基づいて画面を表示するLCD (Liquid Crystal Display) からなるタッチパネル60が接続されており、その他、図示しないが、音を出力するスピーカと、インターネットに接続するための信号線等が接続されている。

【0042】グラフィックチップ56は、VRAM55に格納されている表示用データを先頭アドレスから所定周期で順次読み出し、読み出した表示用データを画像信号に変換してタッチパネル64に出力するようになっている。

【0043】次に、フラッシュROM52の構成を図2ないし図8を参照しながら説明する。図2は、ブロックA、Bのデータ構造を示す図である。

【0044】フラッシュROM52は、データ書換可能な不揮発性の記憶素子からなり、図2に示すように、その記憶領域をデータ初期化単位からなる複数のブロックに論理的に区分し、それらブロックのうち2つのブロックA、Bが、データの書込を行うためのデータ書込用領域、およびブロックの更新時に用いるデータ更新用領域として利用される。なお、フラッシュROM52は、例えば、初期化状態（初期化されてデータが書き込まれていない状態）では記憶領域のデータビットが「1」となるものである。

【0045】ブロックA、Bは、いずれも、ブロックの管理に関する領域管理データを格納するための領域管理部70と、パケットを格納するためのパケット部72とからなっており、一方がデータ書込用領域となるときは、他方がデータ更新用領域となり、逆に、他方がデータ書込用領域となるときは、一方がデータ更新用領域となる。データの書込は、データ書込用領域となるブロックに対して行う。これに対し、ブロックの更新は、データ書込用領域となるブロックのパケット部72にデータを書き込むための領域が不足したときに行うものであって、データ書込用領域となるブロックのパケット部72の有効なパケットのデータを、その配置位置が整順されるように、データ更新用領域となるブロックのパケット部72に書き込むことにより行う。そして、

パケット部72へのデータの書込が完了した時点で、データ更新用領域となっているブロックをデータ書込用領域とし、データ書込用領域となっているブロックをデータ更新用領域とする。

【0046】領域管理部70は、図3に示すように、ブロックの初期化回数を示す初期化回数情報を格納する領域70aと、ブロックの異なる5つの状態を示す領域状態情報1～5を格納する領域70b～70fとからなっており、各領域70a～70fは、フラッシュROM52のデータ読出単位（例えば、4バイト（32ビット））の記憶容量を有している。図3は、領域管理部70のデータ構造を示す図である。

【0047】各領域状態情報1～5は、図4に示すように構成されている。図4は、領域状態情報の内容を示す図である。

【0048】領域状態情報1は、図4の上1段目の項目に示すように、ブロックが初期化されていること（初期化完了状態）を示す情報であって、「INIT」という4バイトの書込情報として領域70bに書き込まれる。また、領域状態情報2は、図4の上2段目の項目に示すように、ブロックへのデータ書込中であること（データ書込中状態）を示す情報であって、「STUP」という4バイトの書込情報として領域70cに書き込まれる。また、領域状態情報3は、図4の上3段目の項目に示すように、ブロックへのデータ書込が完了し、さらにブロックのパケット部72にデータの追記が可能であること（データ書込完了状態）を示す情報であって、「NORM」という4バイトの書込情報として領域70dに書き込まれる。

【0049】領域状態情報4は、図4の上4段目の項目に示すように、ブロックのデータを更新するため、ブロックからのデータ移動中であること（データ移動中）を示す情報であって、「MOVE」という4バイトの書込情報として領域70eに書き込まれる。また、領域状態情報5は、図4の上5段目の項目に示すように、ブロックからのデータ移動が完了したこと（データ移動完了状態）を示す情報であって、「MF IN」という4バイトの書込情報として領域70fに書き込まれる。

【0050】領域状態情報1～5は、その順序で各領域70b～70fに書き込まれるものである。したがって、例えば、領域状態情報3は、領域状態情報1、2が書き込まれていない状態では、書き込まれないし、領域状態情報4、5が書き込まれている状態でも、書き込まれない。なお、領域状態情報1～5がいずれも書き込まれていない状態、または領域状態情報1～5がその順序で書き込まれていない状態は、ブロックが一度も初期化されたことがないもの（未初期化状態）として定義し、取り扱う。

【0051】パケット部72は、フラッシュROM52のデータ読出単位（例えば、4バイト）の倍数からなる

所定容量の記憶領域を有し、図5に示すように、可変長の複数のパケット $P_1 \sim P_n$ を格納可能となっており、各パケット $P_1 \sim P_n$ は、パケットの管理に関する管理データを格納するための管理部80と、記憶すべきデータを格納するためのデータ部82とからなり、パケット部72の先頭アドレスから末尾アドレスに向けて順次生成されていく。図5は、パケット部72のデータ構造を示す図である。

【0052】管理部80は、図6に示すように、パケットの異なる5つの状態を示す状態情報を格納する領域80aと、パケットを識別するためのIDを示すID情報を格納する領域80bと、パケットが生成された順に付す連続番号を示す連続番号情報を格納する領域80cと、データ部82のデータ長を示すデータ長情報を格納する領域80dとからなり、各領域80a～80dは、1バイト（8ビット）の記憶容量を有し、管理部80全体として、フラッシュROM52のデータ読出単位（例えば、4バイト（32ビット））の記憶容量を有している。図6は、管理部80のデータ構造を示す図である。

【0053】状態情報は、上記領域状態情報と異なり、領域80aに格納される1つのデータによりパケットの異なる5つの状態を示すものであって、図7に示すように構成されている。図7は、状態情報の内容を示す図である。

【0054】状態情報は、第1の状態として、パケットへのデータ書込中であること（データ書込中状態）を示し、この場合、「7FH」という1バイトの書込情報として領域80aに書き込まれる。また、第2の状態として、パケットへのデータ書込が完了したこと（データ書込完了状態）を示し、この場合、第1の状態を示す書込情報の「1」の最上位ビットを「0」に設定した「3FH」という1バイトの書込情報として領域80aに書き込まれる。また、第3の状態として、パケットのデータ更新中であること（データ更新中状態）を示し、この場合、第2の状態を示す書込情報の「1」の最上位ビットを「0」に設定した「1FH」という1バイトの書込情報として領域80aに書き込まれる。

【0055】状態情報は、第4の状態として、パケットのデータ更新が完了したこと（データ更新完了状態）を示し、この場合、第3の状態を示す書込情報の「1」の最上位ビットを「0」に設定した「0FH」という1バイトの書込情報として領域80aに書き込まれる。また、第5の状態として、パケットが未使用であること（未使用状態）を示し、この場合、「FFH」という1バイトの書込情報（すなわち、初期化状態のまま）として領域80aに書き込まれる。

【0056】状態情報は、第5の状態、第1の状態、第2の状態、第3の状態および第4の状態の順序で領域80aに書き込まれるものである。したがって、例えば、状態情報が第3の状態であるときは、そのパケットが属

するブロックが初期化されない限り、第1の状態または第2の状態になることはない。

【0057】また、フラッシュROM52には、図8に示すように、ブロックA、Bの領域状態情報の状態に応じてブロックA、Bに対する処理を割り当てた処理定義テーブルが格納されている。図8は、処理定義テーブルのデータ構造を示す図である。

【0058】処理定義テーブルは、図8に示すように構成されている。

【0059】ブロックAの状態がデータ移動中でありかつブロックBの状態が未初期化またはデータ書込中であるときは、ブロックBを初期化する。ブロックAの状態がデータ移動中でありかつブロックBの状態が初期化完了であるときは、ブロックAの有効なパケットのデータをブロックBのパケット部72に書き込むことにより、ブロックAをブロックBに更新する。ブロックAの状態がデータ書込中でありかつブロックBの状態が未初期化若しくは初期化完了であるとき、ブロックAの状態がデータ移動中でありかつブロックBの状態がデータ書込完了であるとき、またはブロックAの状態がデータ移動完了でありかつブロックBの状態が未初期化、初期化完了、データ書込中若しくはデータ書込完了であるときは、ブロックAを初期化する。ブロックAの状態がデータ書込完了でありかつブロックBの状態が未初期化、初期化完了若しくはデータ書込中であるときは、ブロックAをデータ書込用領域とする。

【0060】また、ブロックBの状態がデータ移動中でありかつブロックAの状態が未初期化またはデータ書込中であるときは、ブロックAを初期化する。ブロックBの状態がデータ移動中でありかつブロックAの状態が初期化完了であるときは、ブロックBの有効なパケットのデータをブロックAのパケット部72に書き込むことにより、ブロックBをブロックAに更新する。ブロックBの状態がデータ書込中でありかつブロックAの状態が未初期化若しくは初期化完了であるとき、ブロックBの状態がデータ移動中でありかつブロックAの状態がデータ書込完了であるとき、またはブロックBの状態がデータ移動完了でありかつブロックAの状態が未初期化、初期化完了、データ書込中若しくはデータ書込完了であるときは、ブロックBを初期化する。ブロックBの状態がデータ書込完了でありかつブロックAの状態が未初期化、初期化完了若しくはデータ書込中であるときは、ブロックBをデータ書込用領域とする。

【0061】また、ブロックAの状態が未初期化若しくは初期化完了でありかつブロックBの状態が未初期化または初期化完了であるときは、ブロックA、Bに対する初期化処理を行う。

【0062】一方、図1に戻り、CPU50は、マイクロプロセッシングユニットMPU等からなり、フラッシュROM52の所定領域に格納されている所定のプログ

ラムを起動させ、そのプログラムに従って、図9ないし図13のデータ書込処理、データ読込処理およびブロック更新処理を並列に実行するようになっている。

【0063】まず、データ書込処理を図9を参照しながら説明する。図9は、データ書込処理を示すフローチャートである。

【0064】データ書込処理は、ブロックA、Bに対してデータの新規書込または更新をパケット単位で行うときに実行され、初期状態ではブロックAをデータ書込用領域とし、ブロックAに対してデータの書込を行い、それ以降は処理に応じてデータ書込用領域となるブロックに対してデータの書込を行う処理であって、CPU50において実行されると、図9に示すように、まず、ステップS100に移行する。

【0065】ステップS100では、フラッシュROM52に対してデータの書込を行おうとする他の処理（図9ないし図13のフローチャートに示す処理以外のもの）からのデータ書込要求が、データの新規書込要求であるかパケットのデータの更新要求であるかを判定し、データの新規書込要求であると判定したとき(Yes)は、ステップS101に移行して、ブロックの更新が完了したか否かを判定し、ブロックの更新が完了したと判定されたとき(Yes)は、ステップS102に移行するが、そうでないと判定されたとき(No)は、ブロックの更新が完了するまでステップS101で待機する。

【0066】ステップS102では、記憶すべきデータを格納するためのパケットを生成するための領域がパケット部72に不足しているか否かを判定し、パケットを生成するための十分な領域がパケット部72にあると判定されたとき(No)は、ステップS104に移行する。

【0067】ステップS104では、記憶すべきデータを格納するためのパケットをパケット部72に生成し、ステップS108に移行する。このステップS104では、具体的に、パケット部72にパケットが1つも生成されていないときは、パケット部72の先頭アドレスからパケットを生成し、パケット部72に既にパケットが生成されているときは、パケット部72の先頭アドレスから数えて末尾にあるパケットの末尾アドレスの次のアドレスからパケットを生成する。また、パケットを生成するときは、他のパケットに与えられたIDとは異なるIDを示すID情報、このパケットが生成される順番を示す連続番号、記憶すべきデータのデータ長を示すデータ長情報をそれぞれ管理部80に書き込み、状態情報を第1の状態に設定する。

【0068】ステップS108では、記憶すべきデータを、生成したパケットのデータ部82に書き込み、ステップS110に移行して、生成したパケットの状態情報を第2の状態に設定し、一連の処理を終了する。

【0069】一方、ステップS100で、上記他の処理からのデータ書込要求が、パケットのデータの更新要求

であると判定したとき(No)は、ステップS113に移行して、ブロックの更新が完了したか否かを判定し、ブロックの更新が完了したと判定されたとき(Yes)は、ステップS114に移行するが、そうでないと判定されたとき(No)は、ブロックの更新が完了するまでステップS113で待機する。

【0070】ステップS114では、更新すべきデータを格納するためのパケットを生成するための領域がパケット部72に不足しているか否かを判定し、パケットを生成するための十分な領域がパケット部72にあると判定されたとき(No)は、ステップS116に移行する。

【0071】ステップS116では、更新すべきデータを格納するための新たなパケットをパケット部72に生成し、ステップS118に移行する。このステップS116では、具体的に、パケットを生成するときは、更新前のパケットに与えられたIDと同一のIDを示すID情報、このパケットが生成される順番を示す連続番号、および更新すべきデータのデータ長を示すデータ長情報をそれぞれ管理部80に書き込み、状態情報を第1の状態に設定する。なお、パケットの生成場所については、上記ステップS104と同様である。

【0072】ステップS118では、更新前のパケットの状態情報を第3の状態に設定し、ステップS122に移行して、更新すべきデータを新たなパケットのデータ部82に書き込み、ステップS124に移行して、新たなパケットの状態情報を第2の状態に設定し、ステップS126に移行して、更新前のパケットの状態情報を第4の状態に設定し、一連の処理を終了する。

【0073】一方、ステップS114で、更新すべきデータを格納するためのパケットを生成するための領域がパケット部72に不足していると判定されたとき(Yes)は、ステップS132に移行して、ブロックの更新をすべきブロック更新要求を出力し、ステップS134に移行して、ブロックの更新が完了したか否かを判定し、ブロックの更新が完了したと判定されたとき(Yes)は、ステップS116に移行するが、そうでないと判定されたとき(No)は、ブロックの更新が完了するまでステップS134で待機する。

【0074】一方、ステップS102で、記憶すべきデータを格納するためのパケットを生成するための領域がパケット部72に不足していると判定されたとき(Yes)は、ステップS128に移行して、ブロックの更新をすべきブロック更新要求を出力し、ステップS130に移行して、ブロックの更新が完了したか否かを判定し、ブロックの更新が完了したと判定されたとき(Yes)は、ステップS104に移行するが、そうでないと判定されたとき(No)は、ブロックの更新が完了するまでステップS130で待機する。

【0075】次に、データ読込処理を図10を参照しながら説明する。図10は、データ読込処理を示すフロー

チャートである。

【0076】データ読込処理は、ブロックA、Bに対してデータの読込をバケット単位で行うときに実行される処理であって、CPU50において実行されると、図10に示すように、まず、ステップS200に移行する。

【0077】ステップS200では、バケットの管理部80からデータ長情報を取得し、ステップS202に移行して、取得したデータ長情報に基づいて、フラッシュROM52に対してデータの読込を行おうとする他の処理（図9ないし図13のフローチャートに示す処理以外のもの）から与えられたID情報と一致するID情報を有するバケットの格納位置を特定し、ステップS204に移行する。

【0078】ステップS204では、格納位置を特定したバケットが有効であるか否かを判定し、バケットが有効であると判定されたとき(Yes)は、ステップS206に移行する。このステップS204では、バケットの有効・無効を次のように判定する。バケットの状態情報が第2の状態または第3の状態であるときは、そのバケットが有効であると判定し、バケットの状態情報が第1の状態または第4の状態であるときは、そのバケットが無効であると判定する。

【0079】ステップS206では、格納位置を特定したバケットのデータ部82からデータを読み出し、ステップS208に移行して、格納位置を特定したバケットがバケット部72の先頭アドレスから数えて末尾のものであるか否かを判定し、バケットが末尾のものではないと判定されたとき(No)は、ステップS200に移行するが、そうでないと判定されたとき(Yes)は、一連の処理を終了する。

【0080】一方、ステップS204で、格納位置を特定したバケットが無効であると判定されたとき(No)は、ステップS208に移行する。

【0081】次に、ブロック更新処理を図11を参照しながら説明する。図11は、ブロック更新処理を示すフローチャートである。

【0082】ブロック更新処理は、上記ステップS128で出力されるブロック更新要求を受けてブロックA、Bを更新する処理であって、CPU50において実行されると、図11に示すように、まず、ステップS300に移行する。

【0083】ステップS300では、図8に示す処理定義テーブルを参照して、ブロックA、Bに対する処理を実行し、ステップS302に移行して、上記ステップS128によりブロック更新要求があるか否かを判定し、ブロック更新要求があると判定されたとき(Yes)は、ステップS304に移行する。

【0084】ステップS304では、ブロックAが有効であるか否かを判定し、ブロックAが有効であると判定されたとき(Yes)は、ステップS306に移行する。こ

のステップS304では、ブロックの有効・無効を次のように判定する。ブロックの領域管理部70に領域状態情報3、4が格納されているときは、そのブロックが有効であると判定し、ブロックの領域管理部70に領域状態情報1、2、5が格納されているとき、またはブロックの状態が未初期化であるときは、そのブロックが無効であると判定する。

【0085】ステップS306では、ブロックAの領域管理部70に領域状態情報4を書き込み、ステップS300に移行する。

【0086】一方、ステップS304で、ブロックAが無効であると判定されたとき(No)（この場合、ブロックBが有効であるはずなので、すなわちブロックBが有効であると判定されたとき）は、ステップS308に移行して、ブロックBの領域管理部70に領域状態情報4を書き込み、ステップS300に移行する。

【0087】一方、ステップS302で、上記ステップS128によりブロック更新要求がないと判定されたとき(No)は、ステップS300に移行する。

【0088】次に、上記ステップS300の処理定義テーブルに基づく処理を図12および図13を参照しながら説明する。図12および図13は、ステップS300の処理定義テーブルに基づく処理を示すフローチャートである。

【0089】ステップS300の処理定義テーブルに基づく処理は、上記ステップS300において実行されると、図12に示すように、まず、ステップS400に移行する。

【0090】ステップS400では、ブロックA、Bの領域状態情報を取得し、ステップS402に移行して、図8に示す処理定義テーブルを参照して、取得した領域状態情報からブロックA、Bに対する処理を選択する。

【0091】ステップS402の選択の結果、ブロックAを初期化する処理が選択されたとき(①)は、ステップS404に移行して、ブロックAを初期化し、ステップS406に移行して、ブロックAの初期化回数を「1」加算してブロックAの領域管理部70にその初期化回数情報を書き込み、ステップS408に移行して、ブロックAの領域管理部70に領域状態情報1を書き込み、一連の処理を終了する。

【0092】一方、ステップS402の選択の結果、ブロックBを初期化する処理が選択されたとき(②)は、ステップS410に移行して、ブロックBを初期化し、ステップS412に移行して、ブロックBの初期化回数を「1」加算してブロックBの領域管理部70にその初期化回数情報を書き込み、ステップS414に移行して、ブロックBの領域管理部70に領域状態情報1を書き込み、一連の処理を終了する。

【0093】一方、ステップS402の選択の結果、ブロックBをブロックAに更新する処理が選択されたとき

(㊸)は、ステップS416に移行して、ブロックAの領域管理部70に領域状態情報2を書き込み、ステップS418に移行して、ブロックBの有効なパケットのデータを、その格納位置が整順されるように、ブロックAのパケット部72に書き込み、ステップS420に移行して、ブロックAの領域管理部70に領域状態情報3を書き込み、ステップS422に移行して、ブロックBの領域管理部70に領域状態情報5を書き込み、一連の処理を終了する。

【0094】一方、ステップS402の選択の結果、ブロックAをブロックBに更新する処理が選択されたとき(㊹)は、ステップS424に移行して、ブロックBの領域管理部70に領域状態情報2を書き込み、ステップS426に移行して、ブロックAの有効なパケットのデータを、その格納位置が整順されるように、ブロックBのパケット部72に書き込み、ステップS428に移行して、ブロックBの領域管理部70に領域状態情報3を書き込み、ステップS430に移行して、ブロックAの領域管理部70に領域状態情報5を書き込み、一連の処理を終了する。

【0095】一方、ステップS402の選択の結果、ブロックAをデータ書込用領域とする処理が選択されたとき(㊺)は、ステップS432に移行して、ブロックAをデータ書込領域とし、ステップS433に移行して、ブロック更新要求を解除し、一連の処理を終了する。

【0096】一方、ステップS402の選択の結果、ブロックBをデータ書込用領域とする処理が選択されたとき(㊻)は、ステップS434に移行して、ブロックBをデータ書込領域とし、ステップS435に移行して、ブロック更新要求を解除し、一連の処理を終了する。

【0097】一方、ステップS402の選択の結果、ブロックA、Bに対する初期化処理を行う処理が選択されたとき(㊼)は、ステップS436に移行して、ブロックA、Bを初期化し、ステップS438に移行して、ブロックAの領域管理部70に領域状態情報1を書き込み、一連の処理を終了する。

【0098】一方、ステップS402の選択の結果、上記いずれの処理も選択されないとき(×)は、一連の処理を終了する。

【0099】次に、上記実施の形態の動作を図面を参照しながら説明する。

【0100】まず、ブロックA、Bに対してデータの新規書込をパケット単位で行う場合を説明する。

【0101】初期状態では、ブロックAをデータ書込用領域とし、ブロックAに対してデータの書込が行われる。フラッシュROM52に対してデータの書込を行おうとする他の処理からデータ新規書込要求を受けると、データ書込用領域となるブロックAに対してデータの新規書込が開始される。このとき、データ新規書込要求と併せて、記憶すべきデータおよびID情報が与えられ

る。

【0102】データの新規書込は、次のように行われる。CPU50により、ステップS100～ステップS110を経て、記憶すべきデータを格納するためのパケットがブロックAのパケット部72に生成される。このとき、取得されたID情報、このパケットが生成される順番を示す連続番号、および記憶すべきデータのデータ長を示すデータ長情報がそれぞれ管理部80に書き込まれ、状態情報が第1の状態(データ書込中)に設定される。そして、記憶すべきデータが、生成されたパケットのデータ部82に書き込まれ、生成されたパケットの状態情報が第2の状態(データ書込完了)に設定される。

【0103】データの書込が中断されない通常の状態では、このようにブロックAのパケット部72にデータがパケットとして書き込まれるのであるが、例えば、何らかの原因によりデータの書込が中断された場合は、次のように動作する。

【0104】データの書込が中断された場合は、システムを再起動した後、フラッシュROM52に対してデータの読込を行おうとする他の処理からデータ読込要求を受けると、ここではまだブロックAがデータ書込用領域となっているので、ブロックAに対してデータの読込が開始される。このとき、データ読込要求と併せて、上記他の処理からは上記同一のID情報が与えられたとする。

【0105】データの読込は、次のように行われる。CPU50により、ステップS200、S202を経て、パケットの管理部80からデータ長情報が取得され、取得されたデータ長情報に基づいて、上記他の処理から与えられたID情報と一致するID情報を有するパケットの格納位置が特定される。このとき、格納位置が特定されたパケットは、先程データの書込が中断され、その状態情報が第1の状態となっているので、ステップS204、S208を経て、無効なパケットであると判定され、データの読込が行われない。

【0106】なお、データの書込が中断されず、正常にデータの書込が完了した場合は、データ読込要求を受けると、ステップS200、S202を経て、パケットの管理部80からデータ長情報が取得され、取得されたデータ長情報に基づいて、上記他の処理から与えられたID情報と一致するID情報を有するパケットの格納位置が特定される。このとき、格納位置が特定されたパケットは、その状態情報が第2の状態となっているので、ステップS204、S206を経て、有効なパケットであると判定され、格納位置が特定されたパケットのデータ部82からデータが読み出される。

【0107】また、以上では、ブロックAに対してデータの新規書込を行う場合を説明したが、ブロックBに対してデータの新規書込を行う場合も、上記同様の要領で動作する。

【0108】次に、ブロック A、B に対してデータの更新をバケット単位で行う場合を説明する。なお、ブロック A には、すでにいくつかのバケットが生成されており、ここでは、それらのバケットのうちいずれかのデータを更新する。

【0109】初期状態では、ブロック A をデータ書込用領域とし、ブロック A に対してデータの書込が行われる。フラッシュ ROM 52 に対してデータの更新を行おうとする他の処理からデータ更新要求を受けると、データ書込用領域となるブロック A に対してデータの更新が開始される。このとき、データ更新要求と併せて、更新すべきデータおよび更新対象となるバケットの ID 情報が与えられる。

【0110】データの更新は、次のように行われる。CPU 50 により、ステップ S112～ステップ S126 を経て、更新すべきデータを格納するための新たなバケットがブロック A のバケット部 72 に生成される。このとき、取得された ID 情報、このバケットが生成される順番を示す連続番号、および更新すべきデータのデータ長を示すデータ長情報がそれぞれ管理部 80 に書き込まれ、状態情報が第 1 の状態（データ書込中）に設定される。そして、更新前のバケットの状態情報が第 3 の状態（データ更新中）に設定され、更新すべきデータが新たなバケットのデータ部 82 に書き込まれ、新たなバケットの状態情報が第 2 の状態（データ書込完了）に設定されるとともに更新前のバケットの状態情報が第 4 の状態（データ更新完了）に設定される。

【0111】データの更新が中断されない通常の状態では、このようにブロック A のバケット部 72 にデータがバケットとして書き込み更新されるのであるが、例えば、何らかの原因によりデータの更新が中断された場合は、次のように動作する。

【0112】データの更新が中断された場合は、システムを再起動した後、フラッシュ ROM 52 に対してデータの読込を行おうとする他の処理からデータ読込要求を受けると、ここではまだブロック A がデータ書込用領域となっているので、ブロック A に対してデータの読込が開始される。このとき、データ読込要求と併せて、上記他の処理からは上記同一の ID 情報が与えられたとする。

【0113】データの読込は、次のように行われる。CPU 50 により、ステップ S200、S202 を経て、バケットの管理部 80 からデータ長情報が取得され、取得されたデータ長情報に基づいて、上記他の処理から与えられた ID 情報と一致する ID 情報を有するバケットの格納位置が特定される。このとき、新たなバケットよりも更新前のバケットの方がバケット部 72 の先頭アドレス側に格納されているので、更新前のバケットの格納位置が先に特定されるが、この更新前のバケットは、先程データの更新が中断され、その状態情報が第 3 の状態

となっているので、ステップ S204、S206 を経て、有効なバケットであると判定され、そのバケットのデータ部 82 からデータが読み出される。一方、新たなバケットには、更新すべきデータが途中までしか書き込まれていないので、新たなバケットのデータ部 82 からは、データの読込が行われない。

【0114】なお、データの更新が中断されず、正常にデータの更新が完了した場合は、データ読込要求を受けると、ステップ S200、S202 を経て、バケットの管理部 80 からデータ長情報が取得され、取得されたデータ長情報に基づいて、上記他の処理から与えられた ID 情報と一致する ID 情報を有するバケットの格納位置が特定される。このとき、新たなバケットよりも更新前のバケットの方がバケット部 72 の先頭アドレス側に格納されているので、更新前のバケットの格納位置が先に特定されるが、この更新前のバケットは、その状態情報が第 4 の状態となっているので、ステップ S204、S208 を経て、無効なバケットであると判定され、データの読込が行われない。

【0115】次いで、他の処理から与えられた ID 情報と一致する ID 情報を有する次のバケットの格納位置が特定される。このとき、新たなバケットの格納位置が特定されるが、この新たなバケットは、その状態情報が第 2 の状態となっているので、ステップ S204、S206 を経て、有効なバケットであると判定され、そのバケットのデータ部 82 からデータが読み出される。

【0116】また、以上では、ブロック A に対してデータの更新を行う場合を説明したが、ブロック B に対してデータの更新を行う場合も、上記同様の要領で動作する。

【0117】次に、ブロック A、B を更新する場合を図 14 を参照しながら説明する。図 14 は、ブロック A をブロック B に更新する場合を説明するための図である。

【0118】ブロック A をデータ書込用領域として、ブロック A に対してデータの書込・更新を続けていくと、ブロック A のバケット部 72 は、図 14 に示すように、有効なバケットと無効なバケットが混在する状態となる。図 14 の例では、ID が「01」である有効なバケットと ID が「03」である有効なバケットとの間には、データの更新により無効となった 2 つのバケットが存在する。このようにブロック A のバケット部 72 に無効なバケットが多数存在すると、データの新規書込・更新を行うことができなくなる。

【0119】そこで、フラッシュ ROM 52 に対してデータの更新を行おうとする他の処理からデータ新規書込要求またはデータ更新要求を受けたときに、記憶・更新すべきデータを格納するためのバケットを生成するための領域がブロック A のバケット部 72 に不足していると、ステップ S128 において、ブロック更新要求が出力される。ブロック更新要求を受けると、ブロック A が

らブロックBへの更新が開始される。なお、ブロックBは、すでに初期化されているものとし、すなわち、ブロックBの領域管理部70には、領域状態情報1（初期化完了）が書き込まれている。

【0120】ブロックAからブロックBへのブロックの更新は、次のように行われる。CPU50により、ステップS300～S304を経て、処理定義テーブルが参照されてブロックA、Bに対する処理が実行され、ブロックAが有効であるか否かが判定される。このとき、ブロックAが有効であるので、ステップS306において、ブロックAの領域管理部70に領域状態情報4（データ移動中）が書き込まれる。

【0121】次いで、ブロックAの領域管理部70に領域状態情報4が、ブロックBの領域管理部70に領域状態情報1が書き込まれると、ステップS300、S400、S402、S424～S430を経て、処理定義テーブルが参照されて、ブロックBの領域管理部70に領域状態情報2（データ書込中）が書き込まれ、図14に示すように、ブロックAの有効なパケットのデータが、その格納位置が整順されるように、ブロックBのパケット部72に書き込まれ、ブロックBの領域管理部70に領域状態情報3（データ書込完了）が書き込まれるとともにブロックAの領域管理部70に領域状態情報5（データ移動完了）が書き込まれる。このようにブロックの更新が完了すると、ブロックAは、データ更新用領域となり、ブロックAに代わって今度はブロックBがデータ書込用領域となる。

【0122】そして、ブロックAの領域管理部70に領域状態情報5が、ブロックBの領域管理部70に領域状態情報3が書き込まれると、ステップS300、S400、S402～S408を経て、処理定義テーブルが参照されて、ブロックAが初期化され、ブロックAの初期化回数が「1」加算されてブロックAの領域管理部70にその初期化回数情報が書き込まれ、ブロックAの領域管理部70に領域状態情報1が書き込まれる。

【0123】ブロックの更新が中断されない通常の状態では、このようにブロックAの有効なパケットのデータがブロックBのパケット部72に書き込まれるのであるが、例えば、何らかの原因によりブロックの更新が中断された場合は、次のように動作する。

【0124】まず、データの更新のうちブロックBにデータを書き込んでいる最中に中断された場合は、システムを再起動すると、ブロックAの領域管理部70に領域状態情報4が、ブロックBの領域管理部70に領域状態情報2が書き込まれているので、ステップS300、S400、S402、S410～S414を経て、処理定義テーブルが参照されて、ブロックBが初期化され、ブロックBの初期化回数が「1」加算されてブロックBの領域管理部70にその初期化回数情報が書き込まれ、ブロックBの領域管理部70に領域状態情報1が書き込ま

れる。

【0125】そして、ブロックAの領域管理部70に領域状態情報4が、ブロックBの領域管理部70に領域状態情報1が書き込まれると、ステップS300、S400、S402、S424～S430を経て、処理定義テーブルが参照されて、ブロックBの領域管理部70に領域状態情報2が書き込まれ、図14に示すように、ブロックAの有効なパケットのデータが、その格納位置が整順されるように、ブロックBのパケット部72に書き込み直され、ブロックBの領域管理部70に領域状態情報3が書き込まれるとともにブロックAの領域管理部70に領域状態情報5が書き込まれる。その後は、ブロックAが初期化され、ブロックAの領域管理部70に領域状態情報1が書き込まれる。

【0126】次に、データの更新のうちブロックAを初期化している最中に中断された場合は、システムを再起動すると、ブロックAの領域管理部70に領域状態情報5が、ブロックBの領域管理部70に領域状態情報3が書き込まれているので、ステップS300、S400、S402～S408を経て、処理定義テーブルが参照されて、ブロックAが初期化し直され、ブロックAの初期化回数が「1」加算されてブロックAの領域管理部70にその初期化回数情報が書き込まれ、ブロックAの領域管理部70に領域状態情報1が書き込まれる。

【0127】なお、以上では、ブロックAからブロックBへの更新を行う場合を説明したが、ブロックBからブロックAへの更新を行う場合も、上記同様の要領で動作する。

【0128】このようにして、本実施の形態では、データを記憶するときには、記憶すべきデータを格納するためのパケットをブロックのパケット部72に生成し、データ書込開始時にパケットの状態情報を第1の状態に設定し、パケットのデータ部82に記憶すべきデータを書き込み、データ書込完了時にパケットの状態情報を第2の状態に設定し、状態情報が第2の状態であるパケットを有効なパケットであると判定し、状態情報が第1の状態であるパケットを無効なパケットであると判定するようにした。

【0129】これにより、データの書込が完了するまでは、有効なパケットとして識別されないで、データの書込が完了するまでの間に何らかの原因によりデータの書込が中断されても、データ書込途中のパケットのデータが正しいデータとして取り扱われることはない。また、フラッシュROM52上でのデータの消去は、消去対象となるデータが格納されているパケットの状態情報を第1の状態に設定することにより行えばよいので、フラッシュROM52の初期化回数が少なくなる。したがって、従来に比して、フラッシュROM52の寿命の低下を防止するとともにデータの書換を高速に行うことができ、しかもデータの書込を高い信頼性をもって行うこ

とができる。

【0130】さらに、本実施の形態では、パケットのデータを更新するときは、更新すべきデータを格納するための新たなパケットをブロックのパケット部72に生成し、データ書込開始時にパケットの状態情報を第3の状態に設定するとともに新たなパケットの状態情報を第1の状態に設定し、新たなパケットのデータ部82に更新すべきデータを書き込み、データ書込完了時に新たなパケットの状態情報を第2の状態に設定するとともにパケットの状態情報を第4の状態に設定し、状態情報が第2の状態または第3の状態であるパケットを有効なパケットであると判定し、状態情報が第1の状態または第4の状態であるパケットを無効なパケットであると判定するようにした。

【0131】これにより、データの更新が完了するまでは、新たなパケットは、有効なパケットとして識別されないのので、データの更新が完了するまでの間に何らかの原因によりデータの更新が中断されても、データ書込途中の新たなパケットのデータが正しいデータとして取り扱われることはない。この場合は、更新前のパケットのデータが正しいデータとして取り扱われる。したがって、従来に比して、データの更新を高い信頼性をもって行うことができる。

【0132】さらに、本実施の形態では、データ長情報に基づいて、パケットの格納位置を特定するようにした。

【0133】これにより、パケット部72の記憶容量を超えない任意のデータ長のデータを区分することなく一つのパケットとしてブロックのパケット部72に記憶することができるので、可変長のデータの管理が容易になる。

【0134】さらに、本実施の形態では、ブロックを更新するときは、データ書込開始時にブロックA、Bのうち一方のブロックの領域管理部70に領域状態情報4を書き込むとともに他方のブロックの領域管理部70に領域状態情報2を書き込み、一方のブロックのパケット部72の有効なパケットのデータを他方のブロックのパケット部72に書き込み、データ書込完了時に一方のブロックの領域管理部70に領域状態情報5を書き込むとともに他方のブロックの領域管理部70に領域状態情報3を書き込み、領域管理部70に領域状態情報3、4が書き込まれているブロックを有効なブロックであると判定し、領域管理部70に領域状態情報1、2、5が書き込まれているブロックを無効なブロックであると判定するようにした。

【0135】これにより、ブロックの更新が完了するまでは、更新先となる他方のブロックは、有効なブロックとして識別されないのので、ブロックの更新が完了するまでの間に何らかの原因によりブロックの更新が中断されても、データ書込途中の他方のブロックのデータが正し

いデータとして取り扱われることはない。この場合は、更新前の一方のブロックのデータが正しいデータとして取り扱われる。したがって、データの更新をさらに高い信頼性をもって行うことができる。

【0136】さらに、本実施の形態では、領域管理部70に領域状態情報1、2、5が書き込まれているブロックをデータ更新用領域として用いるようにした。

【0137】これにより、ブロックを更新するたびに新たなブロックがデータ更新用領域となることがないので、フラッシュROM52の記憶領域を有効に利用することができる。

【0138】さらに、本実施の形態では、ブロックA、Bの領域状態情報の状態に応じてブロックA、Bに対する処理を割り当てた処理定義テーブルを参照して、ブロックA、Bに対する処理を実行するようにした。

【0139】これにより、ブロックA、Bの領域状態情報の状態に応じて、ブロックA、Bに対する処理を逐次判定する必要がなくなるので、データの書換に関する処理を簡素化することができる。

【0140】さらに、本実施の形態では、ブロックは、フラッシュROM52のデータ初期化単位からなる。

【0141】これにより、フラッシュROM52の初期化回数を最小限に抑えることができる。

【0142】さらに、本実施の形態では、一方のブロックの領域管理部70に領域管理情報4が書き込まれ、かつ、他方のブロックの領域管理部70に領域管理情報2が書き込まれているときは、他方のブロックを初期化するようにした。

【0143】これにより、何らかの原因により他方のブロックのパケット部72にデータを書き込んでいる最中に中断された場合は、システムを再起動すると、他方のブロックが初期化され、他方のブロックのパケット部72にデータが書き込み直されるので、データの更新をさらに高い信頼性をもって行うことができる。

【0144】さらに、本実施の形態では、一方のブロックの領域管理部70に領域管理情報5が書き込まれ、かつ、他方のブロックの領域管理部70に領域管理情報1、2、3が書き込まれているときは、一方のブロックを初期化するようにした。

【0145】これにより、何らかの原因によりデータの更新のうち一方のブロックを初期化している最中に中断された場合は、システムを再起動すると、一方のブロックが初期化し直されるので、データの更新をさらに高い信頼性をもって行うことができる。

【0146】上記実施の形態において、ブロックA、Bは、請求項1、2、4ないし7記載の記憶領域に対応し、領域状態情報2は、請求項4または5記載の第1の状態に対応し、領域状態情報3は、請求項4記載の第2の状態に対応し、領域状態情報4は、請求項4記載の第3の状態に対応し、領域状態情報5は、請求項4または

5記載の第4の状態に対応している。

【0147】なお、上記実施の形態においては、ブロックを、フラッシュROM52のデータ初期化単位から構成したが、これに限らず、データ初期化単位を2以上組み合わせ構成してもよい。このような構成であっても、フラッシュROM52の初期化回数を最小限に抑えることができる。

【0148】また、上記実施の形態においては、ステップS116からステップS118に移行する間にデータの更新が中断された場合について特に説明しなかったが、こうした場合に対応するため、状態情報が第2、3の状態であるパケットおよび状態情報が第1の状態であるパケットが併存するときは、状態情報が第2、3の状態であるパケットを有効なパケットであると判定するように構成してもよい。

【0149】また、上記実施の形態においては、ステップS124からステップS126に移行する間にデータの更新が中断された場合について特に説明しなかったが、こうした場合に対応するため、状態情報が第3の状態であるパケットおよび状態情報が第2の状態であるパケットが併存するときは、状態情報が第2の状態であるパケットを有効なパケットであると判定するように構成してもよい。

【0150】また、上記実施の形態においては、ステップS306からステップS424に移行する間、またはステップS308からステップS416に移行する間にブロックの更新が中断された場合について特に説明しなかったが、こうした場合に対応するため、領域管理部70に領域状態情報4が書き込まれているブロックおよび領域管理部70に領域状態情報1が書き込まれているブロックが併存するときは、領域管理部70に領域状態情報4が書き込まれているブロックを有効なブロックであると判定するように構成してもよい。

【0151】また、上記実施の形態においては、ステップS428からステップS430に移行する間、またはステップS420からステップS422に移行する間にブロックの更新が中断された場合について特に説明しなかったが、こうした場合に対応するため、領域管理部70に領域状態情報3が書き込まれているブロックおよび領域管理部70に領域状態情報4が書き込まれているブロックが併存するときは、領域管理部70に領域状態情報3が書き込まれているブロックを有効なブロックであると判定するように構成してもよい。

【0152】また、上記実施の形態においては、データ書込用領域としてのブロックおよびデータ更新用領域としてのブロックをそれぞれ1つずつ用いて構成したが、これに限らず、データ更新用領域としてのブロックを1つ、データ書込用領域としてのブロックを複数用いて構成してもよい。この場合は、ブロックの更新を同時に複数行わないことが条件となる。ブロックの更新を同時に

複数行う場合は、データ更新用領域としてのブロックを、記憶領域の更新を同時に行う数だけ用意すればよい。

【0153】また、上記実施の形態において、図9ないし図13のフローチャートに示す動作処理を実行するにあたっては、フラッシュROM52にあらかじめ格納されているプログラムを実行する場合について説明したが、これに限らず、これらの手順を示したプログラムが記録された記録媒体から、そのプログラムをRAM54に読み込んで実行するようにしてもよい。

【0154】ここで、記憶媒体とは、RAM、ROM等の半導体記憶媒体、FD、HD等の磁気記憶型記憶媒体、CD、CDV、LD、DVD等の光学的読取方式記憶媒体、MO等の磁気記憶型／光学的読取方式記憶媒体であって、電子的、磁氣的、光学的等の読み取り方法のいかににかかわらず、コンピュータで読み取り可能な記憶媒体であれば、あらゆる記憶媒体を含むものである。

【0155】また、上記実施の形態においては、本発明に係るフラッシュメモリのデータ管理装置を、フラッシュROM52のデータ初期化単位からなるブロックに、コンピュータ100の設定に必要なデータを可変長のパケット単位で記憶する場合について適用したが、これに限らず、本発明の主旨を逸脱しない範囲で他の場合についても適用可能である。

【0156】

【発明の効果】以上説明したように、本発明に係る請求項1ないし7記載のフラッシュメモリのデータ管理装置によれば、データの書込が完了するまでは、有効なパケットとして識別されないで、データの書込が完了するまでの間に何らかの原因によりデータの書込が中断されても、データ書込途中のパケットのデータが正しいデータとして取り扱われることはない。また、フラッシュメモリ上でのデータの消去は、消去対象となるデータが格納されているパケットの状態情報を第1の状態に設定することにより行えばよいので、フラッシュメモリの初期化回数が少なくなる。したがって、従来に比して、フラッシュメモリの寿命の低下を防止するとともにデータの書換を高速に行うことができ、しかもデータの書込を高い信頼性をもって行うことができるという効果が得られる。

【0157】さらに、本発明に係る請求項2記載のフラッシュメモリのデータ管理装置によれば、データの更新が完了するまでは、新たなパケットは、有効なパケットとして識別されないで、データの更新が完了するまでの間に何らかの原因によりデータの更新が中断されても、データ書込途中の新たなパケットのデータが正しいデータとして取り扱われることはない。したがって、従来に比して、データの更新を高い信頼性をもって行うことができるという効果も得られる。

【0158】さらに、本発明に係る請求項3記載のフラ

ッシュメモリのデータ管理装置によれば、記憶領域の記憶容量を超えない任意のデータ長のデータを区分することなく一つの packets として記憶領域に記憶することができるので、可変長のデータの管理が容易になるという効果も得られる。

【0159】さらに、本発明に係る請求項4記載のフラッシュメモリのデータ管理装置によれば、記憶領域の更新が完了するまでは、更新先となる別の記憶領域は、有効な記憶領域として識別されないのので、記憶領域の更新が完了するまでの間に何らかの原因により記憶領域の更新が中断されても、データ書込途中の更新先となる別の記憶領域のデータが正しいデータとして取り扱われることはない。したがって、データの更新をさらに高い信頼性をもって行うことができるという効果も得られる。

【0160】さらに、本発明に係る請求項5記載のフラッシュメモリのデータ管理装置によれば、記憶領域を更新するたびに新たな記憶領域が別の記憶領域となることがないので、フラッシュメモリの記憶領域を有効に利用することができるという効果も得られる。

【0161】さらに、本発明に係る請求項6記載のフラッシュメモリのデータ管理装置によれば、更新前の記憶領域および更新先となる別の記憶領域の領域状態情報の状態に応じて、更新前の記憶領域および更新先となる別の記憶領域に対する処理を逐次判定する必要がなくなるので、データの書換に関する処理を簡素化することができるという効果も得られる。

【0162】さらに、本発明に係る請求項7記載のフラッシュメモリのデータ管理装置によれば、フラッシュメモリの初期化回数を最小限に抑えることができるという効果も得られる。

【図面の簡単な説明】

【図1】コンピュータ100の構成を示すブロック図である。

【図2】ブロックA、Bのデータ構造を示す図である。

【図3】領域管理部70のデータ構造を示す図である。

【図4】領域状態情報の内容を示す図である。

【図5】パケット部72のデータ構造を示す図である。

【図6】管理部80のデータ構造を示す図である。

【図7】状態情報の内容を示す図である。

【図8】処理定義テーブルのデータ構造を示す図である。

【図9】データ書込処理を示すフローチャートである。

【図10】データ読込処理を示すフローチャートである。

【図11】ブロック更新処理を示すフローチャートである。

【図12】ステップS300の処理定義テーブルに基づく処理を示すフローチャートである。

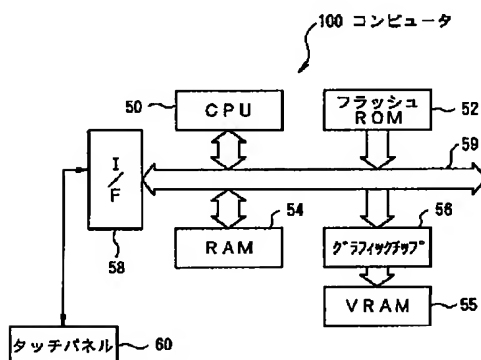
【図13】ステップS300の処理定義テーブルに基づく処理を示すフローチャートである。

【図14】ブロックBに対してデータを更新する場合を説明するための図である。

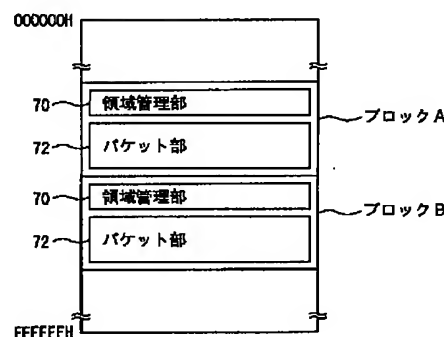
【符号の説明】

100	コンピュータ
50	CPU
52	ROM
54	RAM
55	VRAM
56	グラフィックチップ
57	RTC
58	I/F
59	バス
60	タッチパネル
70	領域管理部
72	パケット部
80	管理部
82	データ部
$P_1 \sim P_n$	パケット

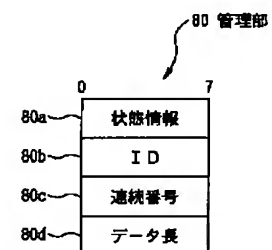
【図1】



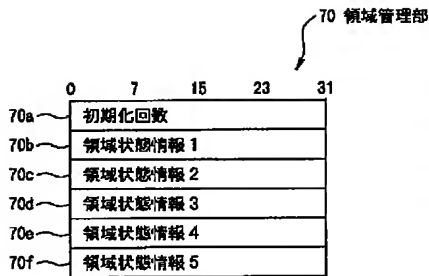
【図2】



【図6】



【図3】



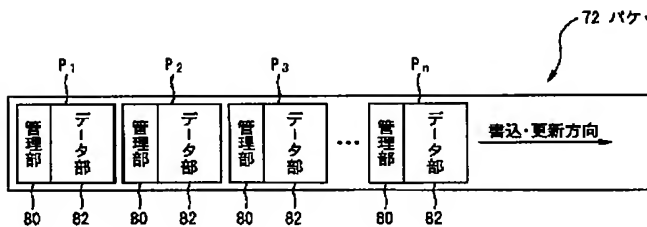
【図4】

領域状態情報	書き情報	状態	意味
領域状態情報 1	INIT	初期化完了	ブロックが初期化されていることを示す。
領域状態情報 2	STUP	データ書き込み中	ブロックへのデータ書き込み中であることを示す。
領域状態情報 3	NORM	データ書き込み完了	ブロックへのデータ書き込みが完了し、データ読み取り可能であることを示す。
領域状態情報 4	MOVE	データ移動中	ブロックからのデータ移動中であることを示す。
領域状態情報 5	MF IN	データ移動完了	ブロックからのデータ移動が完了したことを示す。

上記いずれでもない場合は、「未使用化状態」として定義

【図7】

【図5】

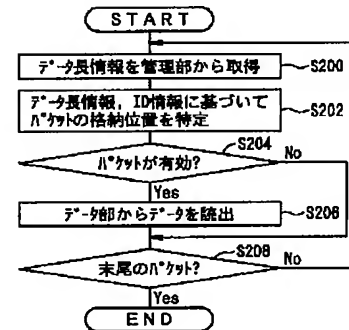


状態情報	書き情報	状態	意味
状態情報 1	7 FH	データ書き込み中	パケットへのデータ書き込み中であることを示す。
状態情報 2	3 FH	データ書き込み完了	パケットへのデータ書き込みが完了したことを示す。
状態情報 3	1 FH	データ更新中	パケットへのデータ更新中であることを示す。
状態情報 4	0 FH	データ更新完了	パケットへのデータ更新が完了したことを示す。
状態情報 5	F FH	未使用	パケットが未使用であることを示す。

【図8】

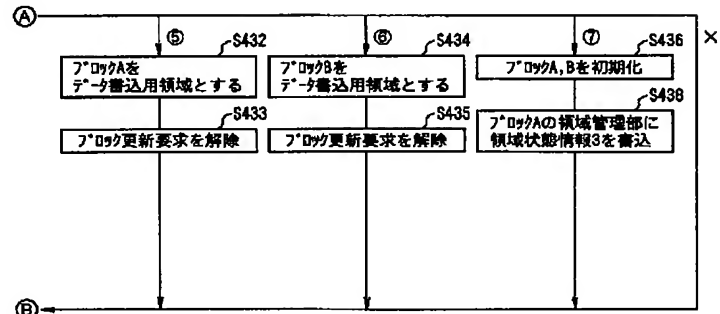
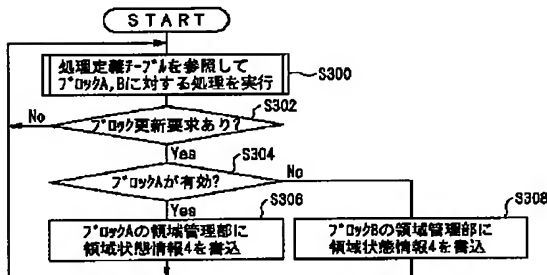
【図10】

ブロックB ブロックA	未初期化	初期化完了	データ書き込み中	データ書き込み完了	データ移動中	データ移動完了
未初期化	⑦ 初期化処理	⑦ 初期化処理	② ブロックBを初期化	⑥ ブロックBをデータ書き込み領域とする	① ブロックAを初期化	② ブロックBを初期化
初期化完了	⑦ 初期化処理	⑦ 初期化処理	② ブロックBを初期化	⑥ ブロックBをデータ書き込み領域とする	③ ブロックBをブロックAに更新	② ブロックBを初期化
データ書き込み中	① ブロックAを初期化	① ブロックAを初期化	×	⑥ ブロックBをデータ書き込み領域とする	① ブロックAを初期化	② ブロックBを初期化
データ書き込み完了	⑤ ブロックAをデータ書き込み領域とする	⑤ ブロックAをデータ書き込み領域とする	×	×	② ブロックBを初期化	② ブロックBを初期化
データ移動中	② ブロックBを初期化	④ ブロックAをブロックBに更新	② ブロックBを初期化	① ブロックAを初期化	×	×
データ移動完了	① ブロックAを初期化	① ブロックAを初期化	① ブロックAを初期化	① ブロックAを初期化	×	×

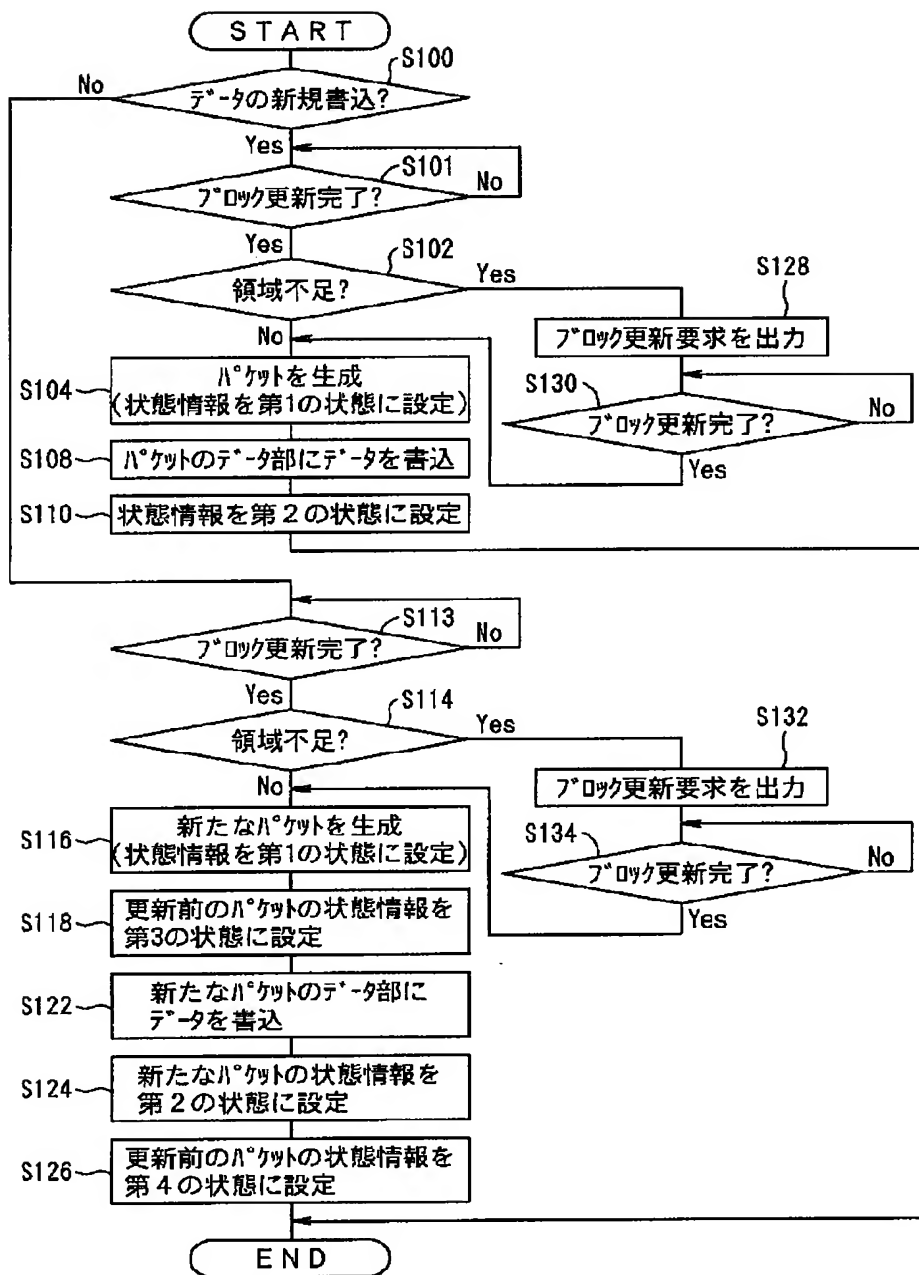


【図11】

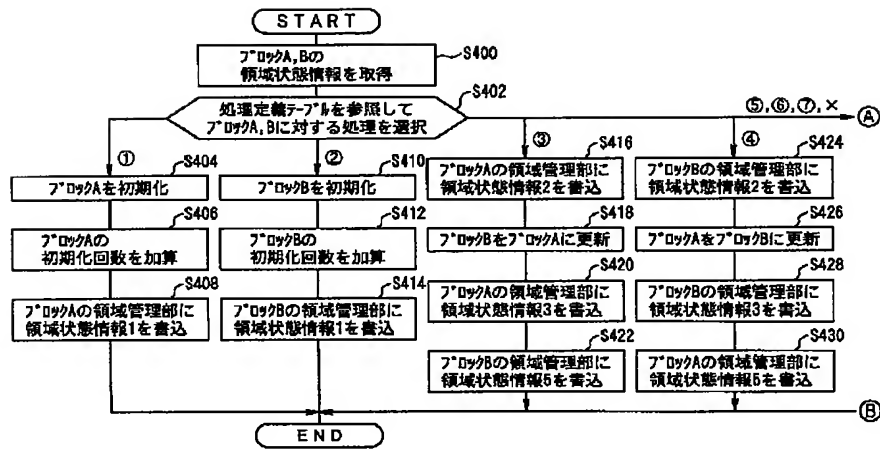
【図13】



【図9】



【図12】



【図14】

